

Remarks

Claims 1-9 are pending. Claims 10-18 have been added. No new matter has been added.

Support for new claims 10-18 can be found throughout the specification of the present application and related applications. For example, the specification discloses at page 20, lines 8-10: persistence information of an object wherein the persistence information accompanies the object and “describes the properties of” the object. Further, related U.S. Patent No. 6,035,119 discloses “a persist operation refers to writing information .. where information relates to the current state of an object ...” at col. 3, lines 38-43.

Summary of Interview

Applicants wish to thank Examiners The T. Ho and St. John Courtenay for the courtesies that they extended during a personal interview held at the U.S. Patent & Trademark Office on August 20, 2003. At the interview, all claims were discussed generally and claims 1, 3, 4, 9 were discussed in detail in view of the applied prior art references Dale (U.S. Patent No. 6,272,673) and Chang (U.S. Patent No. 5,960,436). Specifically, Applicants’ representatives pointed out neither Dale nor Chang teach or suggest instantiating an object on the first machine, outputting the object with persistence information to the second machine and after the object is output from the first machine, where the first processor deletes the instantiation of the object from the first machine as recited in claim 1, for example. Examiner T. Ho alleged that this teaching was provided by Dale (U.S. Patent No. 6,272,673) at col. 7, lines 13-19. Applicants’ representatives indicated that Dale discloses at col. 7, lines 13-19 a server providing an HTML page to a client.

However, it fails to teach or suggest instantiating an object on a first machine, outputting an object with persistence information to a second machine and after the object is output from the first machine, deleting the instantiation of the object from the first machine. It was further pointed out that the disclosure of Dale at col. 13, lines 52-55 that a component is “explicitly destroyed” does not teach or suggest “after the object is output from the first machine”, deleting the instantiation of the object from the first machine. Dale merely discloses deleting in general and does not teach or suggest after the object is output from the first machine, deleting the instantiation of the object from the first machine.

Examiner Ho further asserted that Dale at col. 12, lines 20-23 provided a disclosure of instantiating an object on a first machine, outputting the object with persistence information to a second machine and after the object is output from the first machine, deleting the instantiation of the object from the first machine. Applicants’ representatives indicated that Dale at col. 12, lines 20-23 (in conjunction with Fig. 8b) disclosed an application server 24a (which Examiner Ho equated with the “first machine”) instantiating a component (component 63) on a client 20a (which Examiner Ho equated with the “second machine”) and a second component (component 64) on the application server 24a itself (equated by Examiner Ho to the “first machine”). Applicants’ representatives pointed out that Dale fails to teach or suggest the application server 24a instantiating an object on the application server 24a AND outputting the object with persistence information to a second machine (e.g., the client 20a) AND, after the object is output from the application server 24a, deleting the instantiation of the object from the application server 24a. Thus, Dale merely discloses instantiating a component on a first machine OR

instantiating a component on a second machine. Applicants' representatives pointed out the Dale does not teach or suggest instantiating a component on a first machine, then outputting that object with persistence information to the second machine AND after the object is output from the first machine, deleting the instantiation of the object from the first machine.

Applicants' representatives pointed out that Dale fails to teach or suggest persistence information. During the interview, the Examiners and Applicants' representatives agreed that, based on the discussion, additional areas may be relevant to the claims. The Examiners indicated that a further search would be performed.

Applicants' representatives pointed out that both Dale and Chang fail to teach or suggest modifying the object and outputting the modified object to a second machine. Rather, Chang (relied on in the prior Office Action to provide this teaching) teaches outputting a transaction log but not a modified object. Examiner Ho indicated that further analysis of the Chang reference as well as a further search of the prior art to find pertinent prior art would be performed.

Examiners Courtenay and Ho indicated that the preamble in claim 1 recited elements not present in the body of claim 1. Specifically, the preamble of claim 1 recites "transporting" and "first language" and "second language" which are not recited in the body of claim 1. Pursuant to the Examiners' suggestions, claim 10 has been added and recites these terms of the preamble of claim 1 in the body of new claim 10.

To further prosecution, Applicants have added new claims to more clearly define the invention. In addition, new claims were added reciting an alternative embodiment. However, Applicants do not believe that the presently applied references anticipate or render obvious the

present invention for at least the reasons set forth during the interview and set forth below.

Therefore, the original claims remain in the application.

Rejection over Dale and Admitted Prior Art

Claims 1, 2, and 5-7 were rejected under 35 U.S.C. 103(a) as being unpatentable over U.S. Patent No. 6,272,673, to Dale et al. in view of Admitted Prior Art (APA). This rejection is respectfully traversed.

Claim 1 recites a system for transporting objects between a first and second machine, the system comprising a first processor on the first machine for executing code and instantiating an object on the first machine and an output for outputting the object with persistence information to the second machine wherein after the object is output from the first machine, a first processor deletes the instantiation of the object from the first machine.

The Office Action dated May 20, 2003, continues to equate “the first machine” with the server 24a, Fig. 3 of Dale (and Fig. 8B) and “the second machine” with client 20a of Dale. See May 20, 2003 Office Action at page 3. The May 20, 2003 Office Action also continues to equate the “object” with component 64. See May 20, 2003 Office Action at page 3, section 3, line 9. Dale fails to teach or suggest claim 1. Dale does not teach or suggest the application server 24a (assumed to be the “first machine”) outputting the “component 64” (assumed to be the “object”) with persistence information to client 20a (assumed to be the “second machine”). The application server 24a does not output component 64 at all. Thus, even assuming *arguendo* that the application server 24a is equivalent to the first machine and the component 64 is equivalent to

the object as the Office Action asserts (which it is not), Dale still fails to disclose outputting the component 64 to a *second machine*. The Office Action does not address these deficiencies of Dale.

The APA does not make up for the deficiencies of Dale. The Office Action dated May 20, 2003 cites page 6, lines 3-7 of the specification that discloses: “ASP supports caching data and objects in the server in the Session and Application Objects ... place values in hidden form elements ... the form elements travel with the page to the client ...”. However, the APA does not disclose a first processor on the first machine for executing code and instantiating an object on the first machine and an output for outputting the object with persistence information to the second machine wherein after the object is output from the first machine, a first processor deletes the instantiation of the object from the first machine. Thus, Dale and APA, either alone or in combination, fail to teach or suggest claim 1.

Alternatively, the May 20, 2003 Office Action asserts that the “object” is the “HTML page 62” illustrated in Fig. 8B. See May 20, 2003 Office Action at page 3, section 3, lines 11-12. Even if this assumption were correct, Dale still fails to teach or suggest claim 1. Dale discloses HTML page 62 and components 63, 64 and 65 stored (but not instantiated) in storage facility 27 (Dale at col. 12, lines 10-14 and Fig. 8B). The application server 24a (assumed to be the “first machine” in the Office Action) receives a request for the HTML page 62 (assumed to be the “object” in the Office Action) from the client 20a (assumed to be the “second machine” in the Office Action). The HTML page 62 is provided to the *client 20a*. See Dale at col. 12, lines 15-18. However, Dale fails to disclose a first processor on the first machine (equated with the

application server 24a by the Office Action) for executing code and instantiating an object (equated with the HTML page 62 by the Office Action) on the first machine (the application server 24a), outputting the object (HTML page 62) with persistence information to a second machine (equated with the client 20a by the Office Action) and, after the object (HTML page 62) is output from the first machine (application server 24a), deleting the instantiation of the object (HTML page 62) from the first machine (application server 24a). Specifically, the application server 24a (equated with the “first machine” by the Office Action) merely provides the HTML page 62 (equated with the “object” by the Office Action) to the client 20a (equated with the “second machine” by the Office Action) but does not instantiate the HTML page 62 (equated with the “object”) on *the application server 24a* (the “first machine”). The Office Action fails to address this deficiency in Dale.

Moreover, Dale does not teach or suggest outputting the HTML page 62 with persistence information from the application server 24a (“first machine”) to the client 20a (“second machine”) and after the HTML page 62 (“object”) is output from the application server 24a (“first machine”), deleting the instantiation of the HTML page 62 (“object”) from the application server 24a (“first machine”). The May 20, 2003 Office Action asserts that Dale discloses “outputting the object to the second machine (application server 24a provides the HTML page 62 to the client 20a, lines 17-18 column 12).” (See May 20, 2003, Office Action at page 3, section 3, lines 10-12). However, this assertion is erroneous. Fig. 8B clearly illustrates that the HTML page 62 (equated to the “object” in the Office Action) is output from storage facility 27 (also, HTML page 62 is not instantiated in storage facility 27 – see col. 12, lines 10-14) and not the application

server 24a (equated with the “first machine” in the Office Action). The Office Action fails to address this deficiency in Dale.

Likewise, APA merely discloses caching data and objects in the server in the Session and Application Objects but fails to teach or suggest a first processor on the first machine for executing code and instantiating an object on the first machine and an output for outputting the object with persistence information to the second machine wherein after the object is output from the first machine, a first processor deletes the instantiation of the object from the first machine. As each of Dale and the APA fails to teach or suggest claim 1 and the Examiner has not addressed these missing recitations, there is no showing that the combination would teach or suggest claim 1. Thus, Dale and APA, either alone or in combination, fail to teach or suggest claim 1.

Further, Dale does not disclose after the object is output from the first machine, the first processor deletes the instantiation of the object from the first machine as recited in claim 1. Because Dale does not disclose outputting the object from the first machine at all, Dale clearly does not disclose, *after* the object is output from the first machine, the first processor deletes the instantiation of the object from the first machine. The May 20, 2003, Office Action contends that Dale discloses that the object is “explicitly destroyed.” See May 20, 2003, Office Action at page 3, section 3, lines 13-15. However, Dale merely discloses that a component may be explicitly destroyed (col. 13, line 55) to indicate to other components that the explicitly destroyed component is unregistered and no longer available. It is respectfully pointed out that claim 1 does not recite “explicitly destroying the object” but instead recites “after the object is output from the

first machine, a first processor deletes the instantiation of the object from the first machine.” Dale does not disclose, *after* the object is output from the first machine, the first processor deletes the instantiation of the object from the first machine. Dale does not disclose outputting the object from the first machine at all. The May 20, 2003, Office Action does not address this shortcoming in Dale.

APA merely discloses caching data and objects in the server in the Session and Application Objects and also fails to teach or suggest that, *after* the object is output from the first machine, the first processor deletes the instantiation of the object from the first machine.

The May 20, 2003, Office Action admits that Dale fails to teach or suggest outputting the object with persistence information to the second machine but asserts that it would have been obvious to combine an alleged prior art teaching (APA) wherein form elements travel with a page to a client (specification at page 6, lines 5-6) because “this provide maintaining session and application state information.” (See Office Action dated May 20, 2003 at page 3, section 3, lines 17-21).

It appears that the Office Action attempts to equate “form elements” with “persistence information”. As stated above and in the Appeal Brief filed February 27, 2003, persistence information would be understood by one of skill in the art as information that describes the properties of the object. See Specification at page 20, lines 8-10. The “form elements” as disclosed in the specification at page 6, lines 5-6 refer to an example of an ASP page in which the page contains a form for data entry by a user. The browser may display the page and the user enters data into the form that is posted to the server (specification at page 5, lines 15-23). The

page is managed by placing values in hidden form elements that “travel with the page to the client” and travel with the HTTP POST data back to the server (specification, page 6, lines 5-7). Thus, the form elements do not constitute information describing properties of the object (“persistence information”). Instead, the form elements are data input by the user in the page that travel with the page to the client and server.

Even assuming *arguendo* that the “form elements” are equivalent to “persistence information” as the Office Action has assumed, the proposed combination of Dale and APA is improper because there is insufficient motivation for one of ordinary skill in the art to combine the teachings of Dale with APA.

To establish a *prima facie* case of obviousness, the teaching or suggestion to make the asserted combination and reasonable expectation of success must both be found in the prior art, and not based on applicant’s disclosure. *In re Vaeck*, 947 F.2d 488, 20 USPQ2d 1438 (Fed. Cir. 1991). “To support the conclusion that the claimed invention is directed to obvious subject matter, either the references must expressly or impliedly suggest the claimed invention or the examiner must present a convincing line of reasoning as to why the artisan would have found the claimed invention to have been obvious in light of the teachings of the references.” *Ex parte Clapp*, 227 USPQ 972, 973 (Bd. Pat. App. & Inter. 1985). The mere fact that references can be combined or modified does not render the resultant combination obvious unless the prior art also suggests the desirability of the combination. *In re Mills*, 916 F.2d 680, 16 USPQ2d 1430 (Fed. Cir. 1990).

In the present case, the cited prior art references do not expressly or impliedly suggest the claimed invention. Although the Office Action argues that the references can be combined, this alleged fact, even if true, does not render the resultant combination obvious because none of the cited references suggest the desirability of the combination. *Id.* Moreover, the Office Action fails to present a convincing line of reasoning as to why one of ordinary skill in the art would have found the claimed invention to have been obvious in light of the teachings of the references. The Office Action merely asserts that “it would have been obvious ... because this provides maintaining session and application state information” (Office Action, page 3). However, nowhere does the primary reference (Dale) teach or suggest the need or desirability to utilize persistence information. Moreover, the alleged motivation cited in the Office Action (to provide “maintaining session and application state information”) is likewise not taught or suggested in the Dale reference. In view of the lack of suggestion to modify or combine the cited references, the rejection is improper and should be withdrawn.

Claim 5 recites a data structure for allowing the interchange of objects between a server and a client comprising persistence information associated with a first object representation.

The Office Action concedes that Dale fails to teach or suggest persistence information. (see Office Action dated May 20, 2003 at page 3, section 3, lines 15-16) but asserts that it would have been obvious to combine APA with Dale. However, APA fails to remedy the shortcomings of Dale.

APA merely discloses “form elements” (data input by a user into a form on a page that travels with the page to the client and server – see specification page 5, line 15 – page 6, line 7).

Therefore, neither Dale nor APA teach or suggest persistence information. To establish *prima facie* obviousness of a claimed invention, *all* the claim limitations must be taught or suggested by the prior art. *In re Royka*, 490 F.2d 981, 180 USPQ 580 (CCPA 1974). Because neither Dale nor APA teach or suggest persistence information, not all claim limitations are taught or suggested by the prior art. Thus, the *prima facie* case of obviousness fails.

Even assuming *arguendo* that “form elements” as disclosed in APA is equivalent to “persistence information”, there is no suggestion in the prior art cited to combine the Dale and APA references. To establish *prima facie* obviousness, there must be some reason or motivation to modify or combine the references and the asserted modification or combination must be found in the prior art, and not based on applicant’s disclosure. *In re Vaeck*, 947 F.2d 488, 20 USPQ2d 1438 (Fed. Cir. 1991). Because there is no suggestion in the prior art to modify or combine the references as the Office Action proposes, there is insufficient motivation to one of ordinary skill in the art to modify or combine the references as the Office Action suggests.

Even assuming *arguendo* that the Dale and APA references can be combined, the mere fact that references can be combined or modified does not render the resultant combination obvious unless the prior art also suggests the desirability of the combination. *In re Mills*, 916 F.2d 680, 16 USPQ2d 1430 (Fed. Cir. 1990). In the present case, neither Dale nor APA suggests the desirability of the combination. Rather, only the *Office Action* suggests the desirability of the combination (but nevertheless fails to provide a rationale). This is an exercise of impermissible hindsight. Therefore, even assuming that the Dale and APA references can be combined, this is insufficient to establish *prima facie* obviousness without a suggestion in the prior art (not the

Office Action) that the combination would be desirable. In the absence of such a suggestion, the *prima facie* case of obviousness fails.

Claim 5 recites an object being instantiated on a first machine and being output from the first machine to a second machine, wherein after said object is output from said first machine, the instantiation of said object is deleted from the first machine.

The Office Action equates component 62 with the claimed “object”. However, component 62 is not equivalent to the “object” of claim 5 because component 62 is not disclosed as being instantiated on a first machine and being output from the first machine to a second machine, wherein after component 62 is output from the first machine, the instantiation of component 62 is deleted from the first machine. Instead, Dale discloses component 62 stored, but not instantiated, in storage 27. Dale further provides component 62 to the client 20a (presumably the “first machine” in this example). However, client 20a (as the “first machine”) does not output component 62 to a second machine. Likewise, the application server 24a cannot be the “first machine” because the application server 24a does not instantiate component 62 on itself. Instead, the application server 24a provides component 62 to the client 20a without instantiating on the application server 24a.

The Office Action, in an alternative scheme, equates component 64 with the “object” of claim 5. However, component 64 is also not equivalent to the claimed “object” because component 64 is merely instantiated on the application server 24a (assumed to be the “first machine” by the Office Action). Although claim 5 recites that the object is instantiated on a first machine, claim 5 also recites the object is output from the first machine to a second machine.

Dale fails to teach or suggest the application server 24a outputting component 64 to a second machine. Thus, component 64 cannot be equivalent to the claimed “object” as the Office Action has asserted. Furthermore, Dale does not teach or suggest “after said object is output from said first machine, the instantiation of said object is deleted from the first machine.” The Office Action relies on a general disclosure in Dale regarding destroying a component to provide this teaching (Dale at col. 13, lines 52-55). However, claim 5 does not merely delete or destroy an object or component. Rather, claim 5 recites *after* the object is output from the first machine, the instantiation of the object is deleted from the first machine. Dale fails to provide this teaching or suggestion.

Therefore, it is respectfully submitted that the rejection is improper. The rejection of claim 5 under 35 U.S.C. § 103(a) should be withdrawn.

Claim 6 is similar to claim 1 and is allowable for at least the reasons set forth above for claim 1.

Claims 2 and 7 are dependent from claims 1 and 6, respectively, and are therefore allowable for at least the reasons cited above for claims 1 and 6.

Rejection over Dale, Admitted Prior Art, and Chang

Claims 3 and 8 were rejected under 35 U.S.C. 103(a) as being unpatentable over U.S. Patent No. 6,272,673, to Dale et al. in view of APA and further in view of U.S. Patent No. 5,960,436, to Chang, et al. This rejection is respectfully traversed.

Claims 3 and 8 depend from claims 1 and 6, respectively. As set forth above, Dale or APA, either alone or in combination, does not teach or suggest claim 1 or claim 6. Chang does not make up for the deficiencies of Dale and APA.

Dale and APA fail to teach or suggest claim 1 or claim 6 as set forth above. Chang likewise does not teach or suggest claim 1 or claim 6. Specifically, Chang clearly fails to teach or suggest a first processor on a first machine for executing code and instantiating an object on the first machine, an output for outputting the object with persistence information to a second machine wherein after the object is output from the first machine, the first processor deletes the instantiation of the object from the first machine as recited in claim 1. Nor does the Office Action assert that Chang provides this teaching or suggestion.

Furthermore, as described in the Appeal Brief filed February 27, 2003, Chang teaches that a client requests objects from a server, downloads the objects, stores the objects in a storage location, then disconnects from the server and records and executes transactions locally. After the transactions are recorded locally, the client reconnects with the server and the transactions are sent to the server and replayed at the server to modify objects at the server (col. 2, lines 1-6).

In Chang, a transaction log 500 is maintained at the client (col. 3, lines 63-64). The Transaction file *does not store the actual contents of the objects*, which have been modified by the transaction at the client (col. 4, lines 8-10). The Transaction file (at the client) contains Object Refs for objects in the transaction's read set and write set (col. 3, lines 66-67). The read set 521 consists of object refs but *not the actual objects* (col. 4, lines 39-40). When a client reconnects with the server, it is necessary to replay on the server all of the transactions which

executed, were locally committed, and were logged at the client while it was disconnected (col. 4, lines 31-34). After reconnection of the client to the server, “the *read set of object refs and write set of objects* are sent to the server to be replayed.” Col. 4, lines 48-50. The logged client transactions are replayed (at the server) one by one in sequence. Col. 5, lines 1-2. “Replay” means attempting to commit the final state of the client transaction against the server’s database, as if the transaction had just executed at the server. Col. 4, lines 51-53.

Notably, Chang merely teaches a client performing transactions locally, and then sending the *transactions* to a server to be replayed at the server after reconnection with the server. Objects located at the server are modified based on the transactions being replayed at the server. The read set and write set of objects refs and objects are sent to the server from the client, however the object refs do not contain the actual objects (col. 4, line 40). Therefore, “written back” as the Office Action cites at col. 2, lines 12 of Chang actually refers to the modified objects at the *server* being “written back” and *not* the objects at the client being output. Thus, the modified objects are not disclosed as being output to a second machine. Moreover, Chang does not teach or suggest after the object is output from the first machine, deleting the instantiation of the object from the first machine.

The Office Action asserts that Chang discloses “after the object was being modified by the client computer, it can be sent back to the server for future use by the original client computer of any other client computers that are connected to the server”. See Office Action at page 4, line 22 – page 5, line 2. Claims 3 and 8 do not recite these limitations, however, even assuming *arguendo* that Claims 3 and 8 recited these limitations as the Office Action implies, Chang,

either alone or in combination with Dale or APA, still does not teach or suggest claim 3 or 8. Chang merely sends *transactions* to a server and not modified objects. Thus, contrary to the Office Action's assertions, Chang does not teach or suggest passing modified objects back to the server.

The *prima facie* case of obviousness fails. The rejection of claims 3 and 8 under 35 U.S.C. § 103(a) should be withdrawn.

Rejection over Dale, Admitted Prior Art, Barlow and Chang

Claims 4 and 9 stand rejected under 35 U.S.C. 103(a) as being unpatentable over U.S. Patent No. 6,272,673, to Dale et al. in view of APA and further in view of U.S. Patent No. 6,275,935, to Barlow et al. and U.S. Patent No. 5,960,436, to Chang, et al. This rejection is respectfully traversed.

Claim 4 recites an input in a first machine for receiving persistence information and an event from the second machine, a processor in the first machine for instantiating an object based in part on the persistence information, and an event handler in the first machine for handling the event in combination with modifying the object.

Dale or APA, either alone or in combination, does not teach or suggest a first machine for receiving persistence information and an event from a second machine. Dale, Chang and Barlow fail to teach or suggest persistence information at all. APA merely discloses “form elements” (data input by a user into a form on a page that travels with the page to the client and server – see specification page 5, line 15 – page 6, line 7). Therefore, neither Dale, Chang, Barlow, nor APA

teach or suggest persistence information. To establish *prima facie* obviousness of a claimed invention, *all* the claim limitations must be taught or suggested by the prior art. *In re Royka*, 490 F.2d 981, 180 USPQ 580 (CCPA 1974). Because neither Dale, Chang, Barlow, nor APA teach or suggest persistence information, not all claim limitations are taught or suggested by the prior art. Thus, the *prima facie* case of obviousness fails.

Even assuming *arguendo* that “form elements” as disclosed in APA is equivalent to “persistence information”, there is no suggestion in the prior art cited to combine the Dale, Chang, Barlow and APA references. To establish *prima facie* obviousness, there must be some reason or motivation to modify or combine the references and the asserted modification or combination must be found in the prior art, and not based on applicant’s disclosure. *In re Vaeck*, 947 F.2d 488, 20 USPQ2d 1438 (Fed. Cir. 1991). Because there is no suggestion in the prior art to modify or combine the references as the Office Action proposes, there is insufficient motivation to one of ordinary skill in the art to modify or combine the references as the Office Action suggests.

Even assuming *arguendo* that the Dale, Chang, Barlow and APA references can be combined, the mere fact that references can be combined or modified does not render the resultant combination obvious unless the prior art also suggests the desirability of the combination. *In re Mills*, 916 F.2d 680, 16 USPQ2d 1430 (Fed. Cir. 1990). In the present case, neither Dale, Change, Barlow, nor APA suggests the desirability of the combination. Rather, only the *Office Action* suggests the desirability of the combination (but nevertheless fails to provide a rationale). Therefore, even assuming that the Dale, Chang, Barlow and APA references can be

combined, this is insufficient to establish *prima facie* obviousness without a suggestion in the prior art (not the Office Action) that the combination would be desirable. In the absence of such a suggestion, the *prima facie* case of obviousness fails.

The May 20, 2003, Office Action admits that Dale fails to teach or suggest outputting the modified object but relies on Chang to make up for this deficiency. However, Chang fails to make up for the deficiencies of Dale.

Chang fails to teach or suggest outputting a modified object. Rather, Chang discloses a client downloading an object from a server (col. 2, lines 1-2), storing the downloaded objects (at the client) (col. 2, lines 2-3), executing transactions locally (i.e., at the client) (col. 2, lines 5-6), generating a transaction log locally (i.e., at the client) for logging transactions executing on the client (col. 3, lines 54-57), then, at the server, replaying all of the transactions which were executed and logged on the client (col. 4, lines 31-34). The logged client transactions are replayed (at the server) one by one in sequence (col. 5, lines 1-4). The resultant modified object at the server is written back to the server's database (col. 2, lines 12-13). Thus, the object *at the server* is modified *at the server* and written back into the *server's* database. Notably, contrary to the Office Action's assertions, Chang does not teach or suggest *outputting* the modified object. Thus, neither Dale nor Chang, either alone or in combination, teach or suggest modifying the object and outputting the modified object to the second machine.

Claim 9 is similar to claim 4 and is allowable for at least the reasons set forth above for claim 4.

New claims 10-18 are believed to be allowable over the cited prior art. No new matter has been entered.

Applicants respectfully submit that the instant application is in condition for allowance. If the Examiner feels, however, that further amendment and/or discussion may be helpful in facilitating prosecution of the case, the Examiner is respectfully requested to telephone the undersigned attorney of record at the number appearing below.

Respectfully submitted,

By:



Christopher R. Glembocki
Registration No. 38,800

BANNER & WITCOFF, LTD
1001 G Street, N.W.
Eleventh Floor
Washington, D.C. 20001
(202) 824-3000
Dated: September 22, 2003